

Minimal TSP Tour is coNP-Complete

Marzio De Biasi

marziodebiasi [at] gmail [dot] com

Abstract

The problem of deciding if a Traveling Salesman Problem (TSP) tour is minimal was proved to be coNP-complete by Papadimitriou and Steiglitz. We give an alternative proof based on a polynomial time reduction from 3SAT. Like the original proof, our reduction also shows that given a graph G and an Hamiltonian path of G , it is NP-complete to check if G contains an Hamiltonian cycle (Restricted Hamiltonian Cycle problem).

1 Introduction

The *Traveling Salesman Problem* (TSP) is a well-known problem from graph theory [6],[4]: we are given n cities and a nonnegative integer distance d_{ij} between any two cities i and j (assume that the distances are symmetric, i.e. for all i, j , $d_{ij} = d_{ji}$). We are asked to find the *shortest tour* of the cities, that is a permutation π of $[1..n]$ such that $\sum_{i=1}^n d_{\pi(i), \pi(i+1)}$ (where $\pi(n+1) = \pi(1)$) is as small as possible. Its decision version is the following:

TSPDECISION: If a nonnegative integer bound B (the traveling salesman's "budget") is given along with the distances, does there exist a tour of all the cities having total length no more than B ?

TSPDECISION is NP-complete (we assume that the reader is familiar with the theory of NP-completeness, for a good introduction see [4] or [8]). In [6] two other problems are introduced:

TSPEXACT: Given the distances d_{ij} among the n cities and a nonnegative integer B , is the length of the shortest tour *equal* to B ; and

TSPCOST: Given the distances d_{ij} among the n cities calculate the *length* of the shortest tour.

TSPEXACT is DP-complete (a language L is in the class DP if and only if there are two languages $L_1 \in \text{NP}$ and $L_2 \in \text{coNP}$ and $L = L_1 \cap L_2$); TSPCOST and TSP are both FP^{NP} -complete (FP^{NP} is the class of all functions from strings to strings that can be computed by a polynomial-time Turing machine with a SAT oracle) [6].

Recently a post by Jean Francois Puget: "No, The TSP Isn't NP Complete" and the subsequent reply by Lance Fortnow: "Is Traveling Salesman NP-Complete?" [3] (re-)raised the question of the correct interpretation of the statement "TSP is NP-complete"; indeed, if we are given a tour, checking that

it is the shortest tour seems not to be in NP. A question about the complexity of the following problem:

TSPMINDECISION: Given a set of n cities, the distance between all city pairs and a tour T , is T visiting each city exactly once and is T of minimal length?

was posted on cstheory.stackexchange.com, a question and answer site for professional researchers in theoretical computer science and related fields [2].

We gave an answer with a first sketch of the proof that TSPMINDECISION is coNP-complete, but after formalising and publishing it on arXiv, we discovered that the result is not new and it originally appeared in [5] (see also Section 19.9 in [7]). The proof given by Papadimitriou and Steiglitz is different: they prove that the Restricted Hamiltonian Cycle (RHC) problem is NP-complete starting from an instance of the Hamiltonian cycle problem G and modifying G into a new graph G' that contains an Hamiltonian path, and has an Hamiltonian cycle if and only if the original G has an Hamiltonian cycle. Our alternative proof is a chain of reductions from 3SAT to the problem of finding a tour shorter than a given one, and it may be interesting in and of itself, so we decided not to withdraw the paper.

2 Minimal TSP tour is coNP-complete

Proving that TSPMINDECISION is coNP-complete is equivalent to proving the NP-completeness of the following:

Definition 2.1 (TSPANOTHERTOUR problem).

Instance: A complete graph $G = (V, E)$ with positive integer distances d_{ij} between the nodes, and a simple cycle C that visits all the nodes of G .

Question: Is there a simple cycle D that visits all the nodes of G such the total length of the tour D in G is strictly less than the total of the tour C in G ?

Theorem 2.2. TSPANOTHERTOUR is NP-complete.

Proof. It is easy to see that a valid solution to the problem can be verified in polynomial time: just check if the tour D visits all the cities and if its length is strictly less than the length of the given tour C , so the problem is in NP. To prove its hardness we give a polynomial time reduction from 3SAT; given a 3CNF formula φ with n variables x_1, \dots, x_n and m clauses C_1, \dots, C_m ; we introduce a new dummy variable z and add it to every clause: $(x_{i_1} \vee x_{i_2} \vee x_{i_3} \vee z)$. We obtain a 4CNF formula φ^z that has at least one satisfying assignment (just set $u = \text{true}$). Note that every satisfying assignment of φ^z in which $z = \text{false}$ is also a satisfying assignment of φ .

From φ^z we generate an undirected graph $G = \{V, E\}$ following the same standard transformation used to prove that the Hamiltonian cycle problem is NP-complete: for every clause we add a node c_j , for every variable x_i we add a *diamond-like* component, and we add a directed edge from one of the nodes of the diamond to the node c_j if x_i appears in C_j as a positive literal; a directed edge from c_j to one of the nodes of the diamond if x_i appears in C_j as a negative literal. Starting from the top we can choose to traverse the diamonds

corresponding to variables x_1, x_2, \dots, x_n, u from left to right (i.e. set x_i to *true*) or from right to left (i.e. set x_i to *false*). The resulting directed graph G has an Hamiltonian cycle if and only if the original formula is satisfiable. For the details of the construction see [8] or [1].

We focus on the diamond corresponding to the dummy variable z ; let e_z be the edge that must be traversed if we assign to u the value of *true* (see Figure 1).

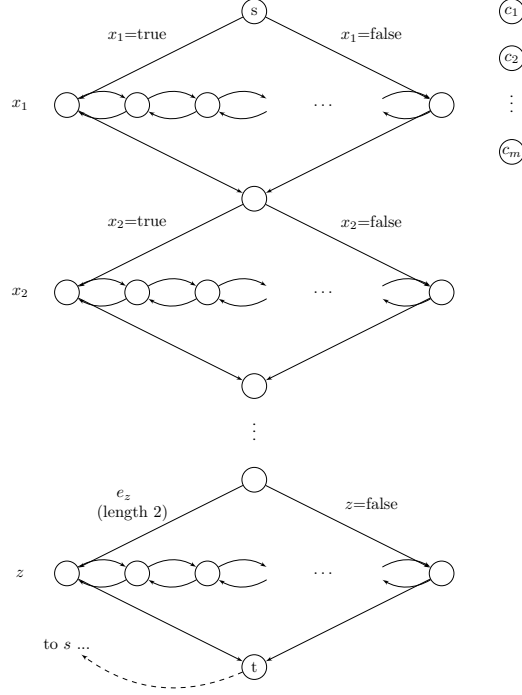


Figure 1: Reduction from 3SAT to directed Hamiltonian cycle.

We can transform G to an undirected graph $G' = \{V', E'\}$ replacing each node $u \in V$ with three linked nodes $u_1, u_2, u_3 \in V'$ and modify the edges according to the standard reduction used to prove the NP-completeness of UNDIRECTED HAMILTONIAN CYCLE from DIRECTED HAMILTONIAN CYCLE [8]: we use u_1 for the incoming edges of u , and u_3 for the outgoing edges, i.e. we replace every directed edge $(u \rightarrow v) \in E$ with $(u_3 \rightarrow v_1) \in E'$. We have G' has an Hamiltonian cycle if and only if G has an Hamiltonian cycle if and only if φ^z is satisfiable.

Finally we transform G' into an instance of TSPANOTHERTOUR assigning length 1 to all edges except edge e_z which has length 2; and we complete the graph adding the missing edges and setting their length to 3.

The dummy variable z guarantees that we can easily find a tour T : just travel the diamonds from left to right without worrying of the clause nodes; when we reach the diamond corresponding to z , traverse it from left to right (i.e. assign to z the value of *true*), and include all the c_j s. By construction the total length of the tour T is exactly $|V'| + 1$: all edges have length 1 except e_u that has length 2.

Another tour D can have a length strictly less than $|V'| + 1$ only if it doesn't use the edge e_u ; so if it exists we can derive a valid satisfying assignment for the original formula φ , indeed by construction φ is satisfiable if and only if there exists a satisfying assignment for φ^z in which $z = \text{false}$. In the opposite direction if there exists a valid satisfying assignment for φ we can easily find a tour D of length $|V'|$: just traverse the diamonds according to the truth values of the variables x_i and traverse the diamond corresponding to z from right to left.

So there is another tour D of total length strictly less than T if and only if the original 3SAT formula φ is satisfiable. □

Hence we have:

Corollary 2.3. *TSPMINDECISION is coNP-complete.*

The reduction used to prove Theorem 2.2 “embeds” the NP-completeness proof of the *Restricted Hamiltonian Cycle problem* (RHC) [7]:

Theorem 2.4. *Given a graph G and an Hamiltonian path in it, it is NP-complete to decide if G contains an Hamiltonian cycle as well.*

Proof. In the reduction above, after the creation of the undirected graph G' , if we remove the edge e_z , we are sure that an Hamiltonian path exists from one endpoint of e_z to the other (just delete e_z from the Hamiltonian cycle that can be constructed setting $z = \text{true}$). An Hamiltonian cycle in $E \setminus \{e_z\}$ *must* use the edge corresponding to $z = \text{false}$, so it exists if and only if the original 3SAT formula φ is satisfiable. □

3 Conclusion

We are optimist: if someone – out there – shouts: “TSP is NP-complete” we are confident that he really means: “The decision version of TSP is NP-complete”; and we hope that, soon or later, someone – out there – will shout “We already know that there is [not] a polynomial time algorithm that solves TSP because P is [not] equal to NP” :-)

Acknowledgements

Thanks to Pálvölgyi Dömötör for the nice hint about Theorem 2.4, and to Marcus Ritt for pointing out the original Papadimitriou and Steiglitz's paper.

References

- [1] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.
- [2] Lance Fortnow. Co-NP-completeness of minimal TSP tour? Theoretical Computer Science – StackExchange, January 2014. <http://cstheory.stackexchange.com/q/20538/3247>.

- [3] Lance Fortnow. Is Traveling Salesman NP-Complete? Computational Complexity blog, January 2014. <http://blog.computationalcomplexity.org/2014/01/is-traveling-salesman-np-complete.html>.
- [4] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [5] C. Papadimitriou and K. Steiglitz. On the complexity of local search for the traveling salesman problem. *SIAM Journal on Computing*, 6(1):76–83, 1977.
- [6] Christos H. Papadimitriou. *Computational complexity*. Academic Internet Publ., 2007.
- [7] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.
- [8] Michael Sipser. *Introduction to the Theory of Computation*. International Thomson Publishing, 1st edition, 1996.